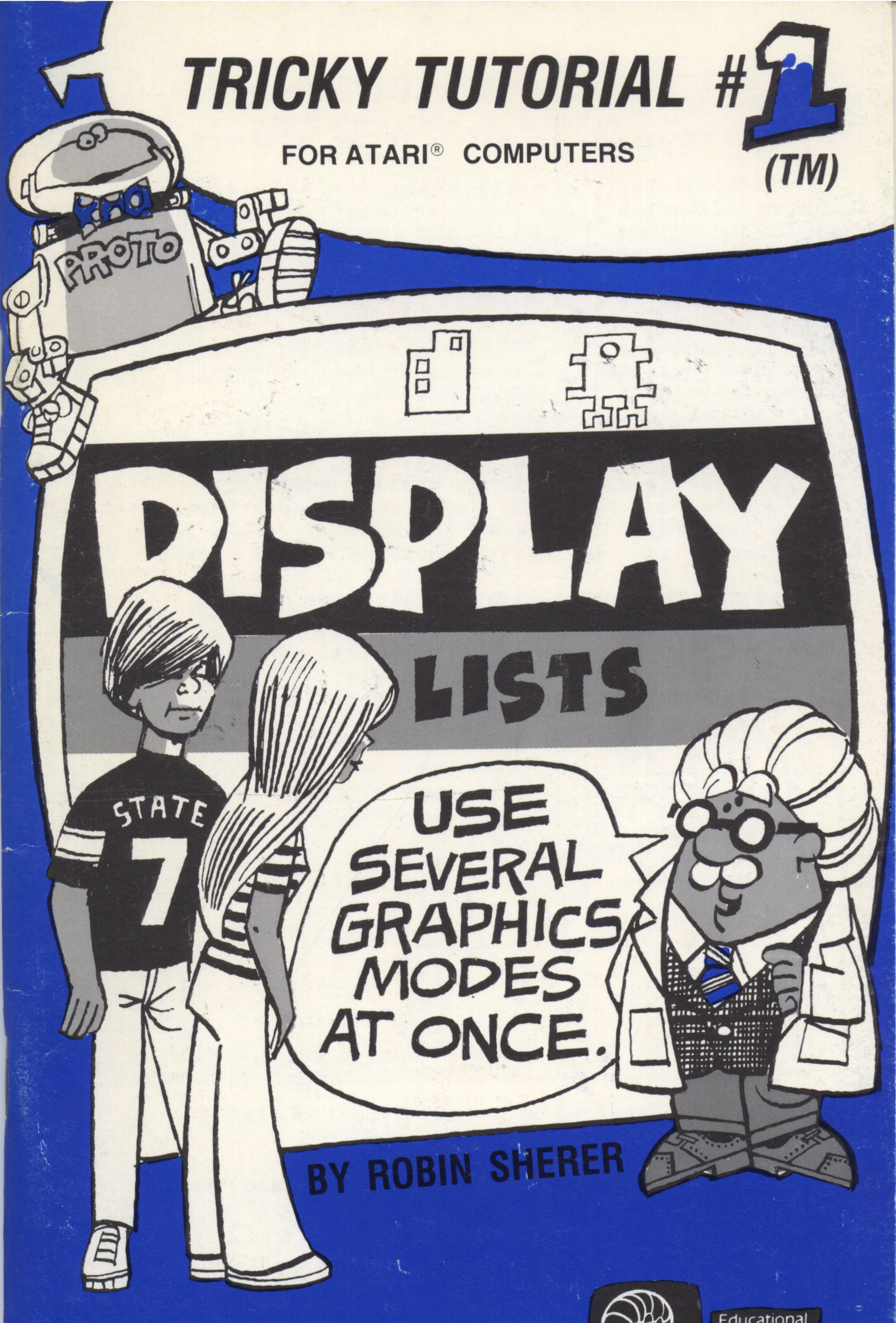
TRICKY TUTORIAL #1

FOR ATARI® COMPUTERS

(TM)

DISPLAY LISTS


PROTO
STATE 7
USE SEVERAL GRAPHICS MODES AT ONCE.

BY ROBIN SHERER

WE MAKE LEARNING TO PROGRAM FUN!

Educational Software inc.
SOQUEL, CA.

ALSO FROM EDUCATIONAL SOFTWARE

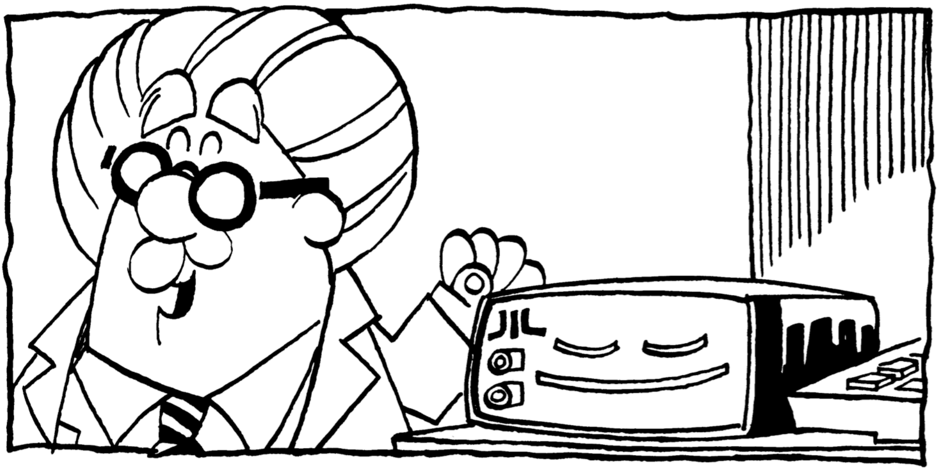
Professor Von Chip's TRICKY TUTORIALS

Here, at last, are the means for knowing, using, and understanding the great power inherent in the ATARI 400/800. Von Chip's TRICKY TUTORIALS take the mystery out of programming by offering several coherent lessons such as:

Tricky Tutorial # 1 -- DISPLAY LISTS
A lesson on mixed graphic screens -- $19.95
Tricky Tutorial # 2 -- HORIZONTAL/VERTICAL SCROLLING
Scroll up, down, sideways, or even diagonally -- $19.95
Tricky Tutorial # 3 -- PAGE FLIPPING
Change screens instantly! -- $19.95
Tricky Tutorial # 4 -- BASICS OF ANIMATION
Beginning animation the easy way. -- $19.95
Tricky Tutorial # 5 -- PLAYER MISSILE GRAPHICS
Create your own video games and animate them. -- $29.95
Tricky Tutorial # 6 -- SOUND and MUSIC
Program your own music, includes PLAYER PIANO -- $19.95

The 1st Six Tricky Tutorials -- Discount Price -- $99.95

Tricky Tutorial # 7 -- DISK UTILITIES
Von Chip's tricks on using disk drives. -- $29.95



Coming Soon from EDUCATIONAL SOFTWARE

Tricky Tutorial # 8 -- CHARACTER GRAPHICS
Make your own character sets and animate them.
Tricky Tutorial # 9 -- GTIA, GRAPHICS 9 TO 11
16 Colors at once!
Tricky Tutorial # 10 -- SOUND EFFECTS
Find that perfect sound effect for your program.
Tricky Tutorial # 11 -- THE MEMORY MAP TUTORIAL
30 lessons demonstrating useful memory locations.

TRICKY TUTORIAL #1

DISPLAY LISTS

by
Robin Sherer

INTRODUCTION

Display List modification is a large and complex subject. If you have seen any of the articles recently published in the hobby magazines on the subject, you know you could never learn to modify your own custom Display Lists.....or could you? What if we let the computer do most of the work?

How to Load

TAPE....

Place the tape in your recorder, label side up. Make sure the tape is rewound, and reset the counter to zero. Push PLAY on the recorder, type CLOAD and press RETURN. When the READY prompt appears, type RUN and press RETURN. If the program won't start to load, try positioning the tape forward or backwards a little. A little trick to find the beginning is to first, turn your volume UP. Then POKE 54018,52 to start the cassette motor. Listen to the "noise" on the tape. When you find the high-pitched, steady tone, you have the beginning of the program. We recommend you write down the number on your recorder's counter as each program starts, this will make it easier to find each part later on. POKE 54018,60 to turn the cassette motor off.

DISK....

To load and run the disk, first turn on your disk drive. When the busy light goes out, place the disk in the drive. Now turn on the computer, with the BASIC Cartridge in place and the program will load each part and run by itself.

Any defective tapes or disks should be returned to:

Educational Software
4565 Cherryvale Ave.
Soquel, CA. 95073

© 1981 EDUCATIONAL SOFTWARE

SO YOU WANT TO LEARN ABOUT DISPLAY LISTS!

To use this program you actually don't need to understand most of the information we will talk about. Feel free to just run the examples and play with creating your own special screens. Later, when you want to understand more about what you are doing, come back and read the booklet and practice modifing the examples. You can't hurt the machine by changing the numbers in the Display List. At the worst, the machine might "go to sleep" (the keyboard will not respond) if a wrong number is POKEd into memory. Then you'll have to press RESET or turn off the machine and reload in the program. This is one reason tape users should always keep track of the number on their recorder's counter for each program so that they can easily reload a program.

This Tutorial doesn't go to the other extreme of dificulty either. MANY details about DL's are not mentioned. The lesson is designed to explain all the basic's of the subject as well as offer examples already typed in for you! (I've got to keep those two recycled beer cans,Prototype & Mototype busy doing something!) If you don't practice modifing the examples with your own ideas, DL'S probably won't make sense to you.

NOTE - From this point on we are going to refer to Display Lists as DL so please don't get confused.

What is a DL?

Why did you buy this program? A surprising number of people buy my tutorials because they want to know how to use the special tricks that the ATARI can do, not really understanding what these tricks are. With this in mind, we'll start out this lesson by explaining some basics of what a Display List is. EXAMPLE 9 shows one suggested use, but you can come up with you own unique ideas!

The ATARI has a special chip inside it to take the information from within its memory and put it on the screen. This chip is called the ANTIC. This chip is actually a microprocessor by itself and thus has a set of instructions to program it....just like the main microprocessor in the ATARI, the 6502. The difference is we don't use BASIC or Assembly language to program it. Now stop and think a moment, I said TWO microprocessors. This gives the machine much more capability than an Apple or Pet (go brag to your friends for a moment then come back........)

The Display List is what the ANTIC looks at to tell the machine how to treat the screen (plus other things not mentioned in this lesson):

The Display List is what the ANTIC looks at to tell the machine how to treat the screen (plus other things not mentioned in this lesson):

1)What Graphics and text modes to put on the screen.

2)Where to get the data to put into these mode areas.

Now don't worry that the subject is too complex. There are only four types of instructions, and these are POKEd into memory using a few basic statements. Let's review - The DL says something to the computer that means: "Put on the screen some Graphic Mode 0, now some Mode 7, then some Mode 0 again, and finally some Mode 2. Oh, by the way, get the data from this location in memory".

We have to discuss the meaning of a few words before we can go on. You already have used Graphics Modes 0 to 8 as described in the BASIC Manual that came with the computer. If you look at Figure 1 you will see a chart showing something called O.S. Modes. The Display List instructions allow the use of these Operating Systems Graphics Modes from 2 to 15. Notice that Graphics Mode 0 is O.S. Mode 2 and Graphics Mode 8 is O.S. Mode 15. In the middle are some new modes you can use from BASIC that ATARI didn't mention in their BASIC book. Uses for these new modes will be mentioned later. So just remember when we say Graphic Mode we mean the one you normally use from BASIC and the O.S.(Operating System) Mode is the one you place in the Display List(through the use of our examples).

O.S MODE [# for DL]	Basic GR. mode	# of Pixel lines per mode line	# of Colors	Bytes of Memory Used Per Line	Text or Plot Mode	Additional
0	-	-	-	-	blank line	In the DL:
1	-	-	-	-	jump	16=2 blank lines
2	0	8	2	40	text	32=3 " "
3	new	10	2	40	text	48=4 " "
4	new	8	4	40	text	68=5 " "
5	new	16	4	40	text	80=6 " "
6	1	8	5	20	text	96=7 " "
7	2	16	5	20	text	112=8 " "
8	3	8	4	10	plot	65=jump & wait
9	4	4	2	10	plot	
10	5	4	4	20	plot	
11	6	2	2	20	plot	
12	new	1	2	20	plot	
13	7	2	4	40	plot	
14	new	1	4	40	plot	
15	8	1	2	40	plot	

FIGURE 1

Here is another idea to learn about. We are going to talk about both Mode Lines and Pixel Lines. To see a Pixel Line, simply look closely at the screen of your TV set. Those tiny dots you see are pixels, so a row of them across the screen is called a Pixel Line. The Mode Lines are harder to understand. If you look at some regular Mode 0 text(small white letters with blue background) you can see that the letters are made up of eight rows of pixels (including the space at the top and bottom). Now look at the chart for GR. mode 0. Under the value called "Pixel Lines per mode line", it says "8", meaning a character in GR.0 takes 8 rows of pixels. You'll use this value later to help yourself in setting up the DL. To review (see Fig.1), a mode line of GR. 8 takes only 1 row of pixels, and O.S. mode 5 (no equivalent GR. mode) takes 16 rows of Pixel Lines, thus, adding GR.5 to your DL fills up the screen 16 times faster than a GR. mode 8 line (O.S. mode 15....look at the chart!). When you input a mode into the Display List in the examples, you must remember if you take out a mode line that is made up of many Pixel Lines, you should replace it with several Mode Lines that use less pixels per line. This will keep the total number of Pixel Lines drawn on the screen the same. The standard number of Pixel Lines drawn on the screen is 192. If you put too few Pixel Lines on the screen, then the picture will "shrink" with just black on the bottom. Likewise, too many will place part of the picture off the screen and may cause the picture to "roll", which can NOT be stopped with the Vertical Hold controls on your TV. If you practice it will all make sense!

NO PROTO, VERTICAL ROLL NOT ROCK & ROLL!

EXAMPLE 1

If you haven't already, please RUN Example 1. This little program allows you to look at any of the standard GR. mode DL's, including the new (to the United States) GTIA modes 9, 10, & 11. Let's try inputting a 0. The numbers that result from the example are those of a standard small text screen. You read the numbers a row at a time.

EXAMPLE OF
GRAPHICS MODE 0

```
112 <---8 blank lines
112 <---8 blank lines
112 <---8 blank lines

 66 <---LMS (includes first line of screen;
                ie. 64(LMS) + 2(O.S. Mode)

 64 <---Low Byte\
                   combine to get location of Data
                   for upper left corner of screen
156 <---High Byte/

  2 <---O.S. mode line (same as GR.0)
  2 <---O.S. mode line
  2 <---O.S. mode line
  2 <---(and all of the other 2's) O.S. mode line

 65 <---Says end of Display List, go to the
        Display List at next location

 32 <---\
         Combine=Low Byte + High Byte * 256
156 <---/
```

FIGURE 2

The first three numbers are 112. Look at Figure 2. You'll see that "112" tells the computer to place 8 blank lines on the screen, starting at the top of your set. Three of these commands causes 24 blank lines at the top of the screen. This number of blank lines is standard. They allow for the difference in individual TV sets, called overscan, so that your whole picture can be seen.

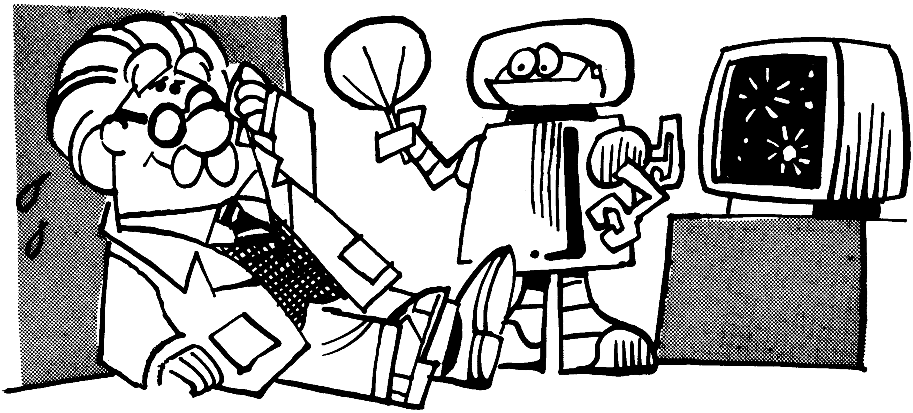
The next number, "66", is the LMS instruction. LMS means Load Memory Scan. LMS tells the computer to "Go find the data for the following modes starting at the address in memory in the next two numbers". We will always use for this number(LMS) the value of 64+the O.S. Mode number.....this means for our GR.0 DL we need 64+2, or 66. Remember that the chart gives GR.0 = O.S. Mode 2. The next two numbers will be the location of the start of the data that is to be put on the screen. To find the value for your machine (if your interested) take the second number of the address(high byte), multiply it by 256 and add the first number (low byte). For example, if the high part of the number was 132 and the low part was 192 then the start of the screen data would be at 256*132+192 which equals 33984. If you POKEd a number into this location that was the value of a letter, number, or Graphics character, you would see it appear at the upper left corner of your TV screen. In this example, POKE 33984,33, an "A" will appear. This is because a 33 in the screen data means "put an A on the screen at this location."

NOTE!

I want to explain an important point about the "Screen Data". The information (data) that is put on the screen is normally stored all in one place in memory, and in the same form. It is the type of mode line you write the data on that determines if it will be large or small text, or graphics pixels. Later some of our examples LIST the program to get data to "flow" through the area where the screen data is stored. Please look closely at the way the same information is INTERPRETED differently as it crosses through the different modes. This concept is IMPORTANT!

Next in the DL come several # 2's. These numbers, we already mentioned, are the O.S. modes that you want. Since we are in GR.0, the DL needs 2's. For each of the 2's(including the one "hidden" in the LMS number before), the computer will put a GR. 0 mode line (8 pixels high!!) across the screen. Don't forget the mode line in the LMS instruction!

Finally comes the number 65. Every DL we will do has one. The chart gives the meaning of the number 65 as "go back to the Display List at the following address and wait for the next time the screen is drawn". Don't worry about this except to include the 65 at the end of your DL. Leave whatever numbers come after this alone. You could actually flip to a completely new DL and it's data by just changing these numbers.



WOW...THAT WAS A LOT!

Sure it was, but that's about it for the technical talk. Now we can have some fun! Input other numbers (0 to 11) into Example 1. Notice that the DL gets longer as you use modes with less Pixel Lines per Mode Line; ie. for a mode like 7 where a mode line is only 2 pixels high, the DL must have a lot of number 13's in it to fill up the screen. Input 8, then look in the middle of the DL. See the 79. There is already a 79 in the top of the DL for the LMS number. Recall that 79 =64+15(the O.S.mode for GR. 8), but why a second LMS number and a second set of addresses for the screen data? The reason is that the DL can only locate up to 4k (4096 bytes) of data. Then it gets lost! The second LMS comes at the point where it ran out of data and says "Here's where to go find some more data to fill the remainder of screen". The second LMS is due to a hardware limit. You will have to make careful calculations to work with it, but only if you use a DL that requires more than 4k on screen data!

```
1000 GRAPHICS 17:? #6;"  EXAMPLE 1"
1010 FOR W=1 TO 230:POKE 710,W
1020 FOR ZR=1 TO 10:NEXT ZR
1030 NEXT W
1100 GRAPHICS 0
1110 DIM NUM(12):DIM A$(1):DIM M(212)
1112 FOR I=1 TO 12:READ X:NUM(I)=X:NEXT I
1114 DATA 32,34,24,34,54,54,94,94,176,202,202
,202
1120 TRAP 40000:TRAP 1120:? "↰":POSITION 1,5:
? "WHICH GRAPHICS MODE WOULD YOU LIKE TO  SEE
THE DISPLAY  LIST FOR";
1125 POKE 710,0
1130 INPUT MODE
1140 IF MODE>11 OR MODE<0 THEN 1165
1150 GOTO 1220
1165 GRAPHICS 17:POKE 708,52
1170 POSITION 1,5:? #6;"MODES 0 TO 11 ONLY":S
OUND 0,120,8,8:FOR W=1 TO 500:NEXT W:SOUND 0,
0,0,0
```

```
1180 FOR W=1 TO 500:NEXT W:GOTO 1120
1220 USE=NUM(MODE+1)
1232 GRAPHICS MODE
1234 NON=PEEK(559):POKE 559,0
1240 J=1
1245 DL=PEEK(560)+256*PEEK(561)
1250 M(J)=PEEK(DL+J-1):J=J+1
1252 IF J<USE+1 THEN 1250
1254 GRAPHICS 0:K=1:POKE 559,NON
1260 POSITION 8,0:? "DISPLAY LIST FOR MODE ";
MODE:POSITION 1,1:? "------------------------
-------------"
1270 FOR J=2 TO 22
1280 FOR I=1 TO 40 STEP 4
1290 POSITION I,J:? M(K):K=K+1:IF K>USE THEN
1310
1300 NEXT I:NEXT J
1310 POSITION 2,23:? "WOULD YOU LIKE TO SEE A
NOTHER ";:INPUT A$
1320 IF A$="Y" THEN 1120
1330 IF A$="N" THEN POKE 764,12:RUN "D:EX2"
1340 GOTO 1260
```

EXAMPLE 2 (at last)

If you are still in example 1 type in "n" and press RETURN. Example 2 is for you to practice changing a Display List. The numbers that come up in a data statement on the screen are there for you to edit using the cursor controls. Numbers can be added, changed, or deleted. When ready to see what the screen will look like with your new numbers, press RETURN. In case you don't understand what to do we have already changed some 2's to 4's, so just pressing RETURN will get a custom screen. Now you can see why ATARI didn't tell you about some of the special modes. Mode 4 is a multicolored text mode designed for use with character graphics, but it sure looks funny! It also is difficult to use, so these new modes will not be explained here. Try them on your own, or see ATARI publications.

Play with this example awhile. Try different numbers in the list. Remember (I keep repeating myself), the numbers in the DL are the O.S. Mode numbers from 2 to 15. This is designed as a simple example, so don't get frustrated that you have to keep starting over with the same list. Later examples will remember your changes and have more room in the DATA statements for numbers. ANYTIME YOU WANT TO CHECK THE MODES YOU HAVE PLACED ON THE SCREEN, THE EASIEST WAY TO DO IT IS TO BREAK THE PROGRAM AND TYPE LIST. THE LISTING WILL FLOW ACROSS THE MODES SHOWING YOU WHERE TEXT AND GRAPHICS ARE.

```
2 GRAPHICS 17:? #6;"    EXAMPLE 2"
3 FOR W=1 TO 230:POKE 710,W
4 FOR ZR=1 TO 10:NEXT ZR
5 NEXT W
7 GRAPHICS 0:DIM ZX$(135)
10 DL=PEEK(560)+256*PEEK(561)
15 A=PEEK(DL+4):B=PEEK(DL+5)
20 GRAPHICS 0:? "THIS EXAMPLE ALLOWS YOU TO C
HANGE A    MODE 0 DISPLAY LIST!"
30 ? "PRESS START TO BEGIN"
55 DATA 112,112,112,66,64,156,2,2,2,2,2,2,2,2
,2,4,4,4,4,4,2,2,2,2,2,2,2,2,2,65,32,156,0,0,
0,0
62 POKE 53279,8
65 Z=PEEK(53279)
70 IF Z=6 THEN 100
80 GOTO 65
100 TRAP 40000:? "↰":POSITION 2,5
105 ? "55 DATA 112,112,112,66,";A;",";B;",2,2
,2,2,2,2,2,2,2,4,4,4,4,4,2,2,2,2,2,2,2,2,2,65
,32,156,0,0,0,0"
110 POSITION 2,15:? "CHANGE THE VALUES YOU WA
NT BY USING    THE CURSOR CONTROLS-THEN PRESS
RETURN TO SEE NEW ";
115 ? "SCREEN"
120 POSITION 0,1
125 INPUT ZX$
130 POSITION 0,5
140 POKE 842,13
142 POSITION 2,12:? "CONT"
144 POSITION 0,3:STOP
150 POKE 842,12
200 RESTORE 55
210 I=0
215 TRAP 100
220 READ G:POKE DL+I,G:I=I+1:IF G=0 THEN 235
230 GOTO 220
235 FOR W=1 TO 900:NEXT W:LIST
240 ? "PRESS OPTION TO GO ON":? "PRESS START
TO DO ANOTHER SCREEN"
245 TRAP 100
250 POKE 53279,8
260 Z=PEEK(53279)
270 IF Z=6 THEN 57
280 IF Z=3 THEN RUN "D:EX3"
300 GOTO 260
```

EXAMPLE 3

This is a small subroutine for you to use in your own programs. The numbers for the DATA statement (the DL) can come from EX. 2 (or better yet EX. 10). Some of the lines in this example we added to make this Tutorial flow smoothly, so after you copy the example, delete all but the call to the subroutine in line 700 plus the subroutine itself, lines 10000-10040 . Here's how the routine works:

```
10010 DL=PEEK(560)+256*PEEK(561)
```

Line 10010 - puts together the two parts of the address stored in 560 & 561. These two locations combine to give you the start of the Display List. Since this varies, you have to look at these locations each time you do a Graphics call, like GR. 0 or GR. 7. The Graphics call will automatically set up a DL and screen data area within memory.

I'm confused!

Right! We need another chart (next page, Figure 3). On this chart are every standard Display List. Look at the first one, for Graphics 0. The Display List starts about 1k from the top of memory with the DL itself. Then, moving up in memory comes the 960 bytes of data that the screen needs. Example 2 calls GR. 0, then looks at 560 & 561 and changes the DL. The 960 bytes of screen data is enough for the modifications you can make with the small DATA statement we have given you so far. Later, when we give you four DATA statements you may want to start with a Graphics Mode that gives you more room like 5,7,even 8. The key is to use only as much of memory as you need. The real way to calculate what mode to start with is to add up how many lines of each mode you are using, multiply each of these by the number of bytes per mode line from FIG.1 , and get a total for them all. For example:

```
    13 lines of GR. 8 * 40 bytes per line
  +  8 lines of GR. 2 * 20 bytes per line
  + 25 GR. 7 lines * 40 bytes per line
    ------------------------------------
    equals:1680 bytes needed for DATA
       and:191 pixel lines used(close enough to 192!)
```

If you allow another 70 or so bytes for the DL itself, you get a minimum of 1750 bytes needed for your custom DL. By calling a standard GR. 6 (see chart), 2048 bytes would be set up which is enough. At this point (after the call), you would look at 560 & 561 to get the address of the start of the DL. By the way, the estimate of 70 additional bytes for the DL comes from adding up the number of mode lines you are going to call (46 in above example), plus some extras for the blank lines, the LMS bytes, and the 65 & address at the end of every DL.

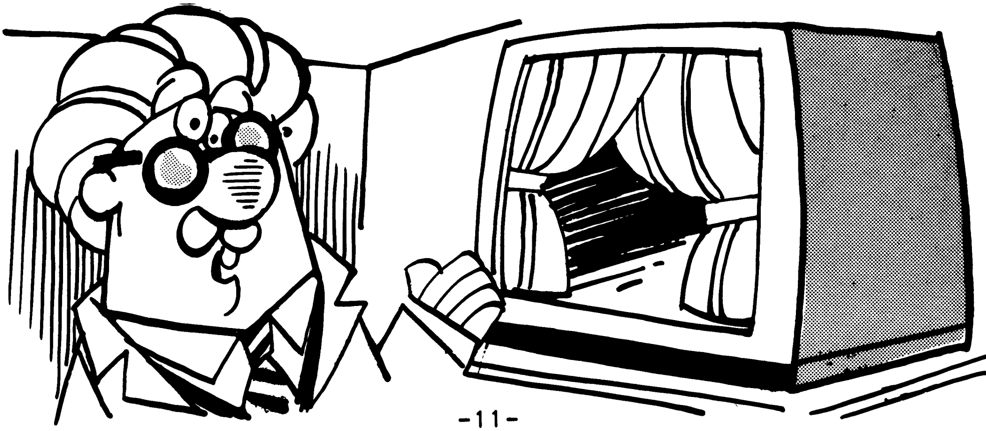
Let's finish this example and then explain the chart.

```
10011 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
```

Line 10011 - saves the location where the screen's data starts, which always comes at the 5th and 6th value into the DL(right after the LMS).

```
10012 I=1
10013 READ A:IF A=0 THEN 10040
10014 POKE (DL+I-1),A
10015 IF I=5 THEN POKE DL+4,Z1
10016 IF I=6 THEN POKE DL+5,Z2
10020 I=I+1:GOTO 10013
10030 DATA 112,112,112,66,64,156,2,2,2,5,5,5,5,2,2,
           4,4,4,4,4,65,32,124,0,0,0
```

Lines 10012 to 10030 - We now set a counter, and read a value; if it = 0 we're all done; if not, we POKE it into the first location in the DL(thus changing the DL if the number is different). We then test for the 5th and 6th value, and if we just POKEd in these values from the DATA statement, we POKE in another value from line 10011. This is because these two numbers depend on your memory size, so the numbers in the data statement are dummys waiting for these correct numbers to be PEEKed from the DL and then POKEd right back in again(sure it could be done other ways....this is my way). This simple loop continues until the program reads in a 0 from the DATA statement then stops. Notice whenever you change a DL this way it changes down the screen just like a curtainfalling. Neat!



DISPLAY LIST FOR GRAPHICS 0 to 4+16

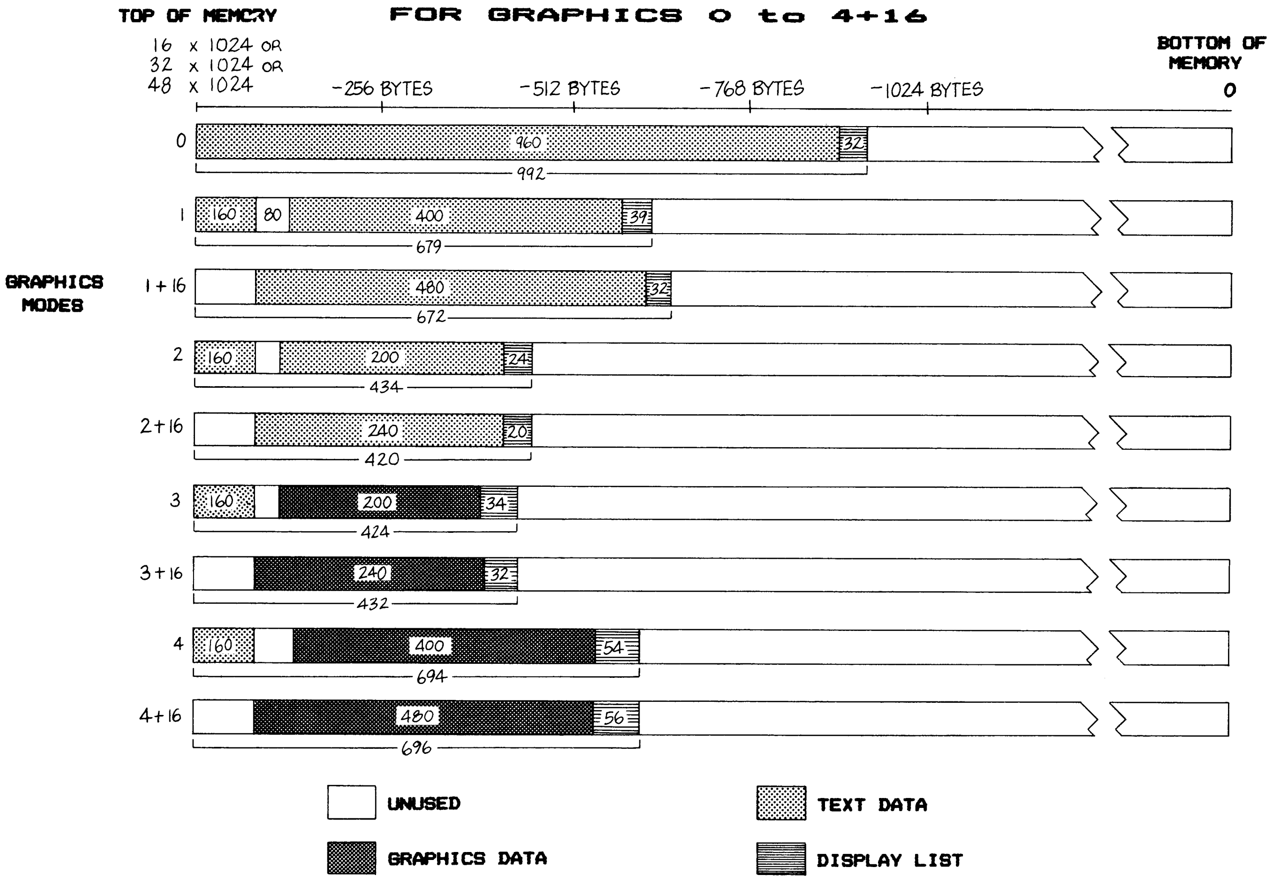


FIGURE 3A (NOT TO SCALE)

DISPLAY LIST
FOR GRAPHICS 5 to 8+16

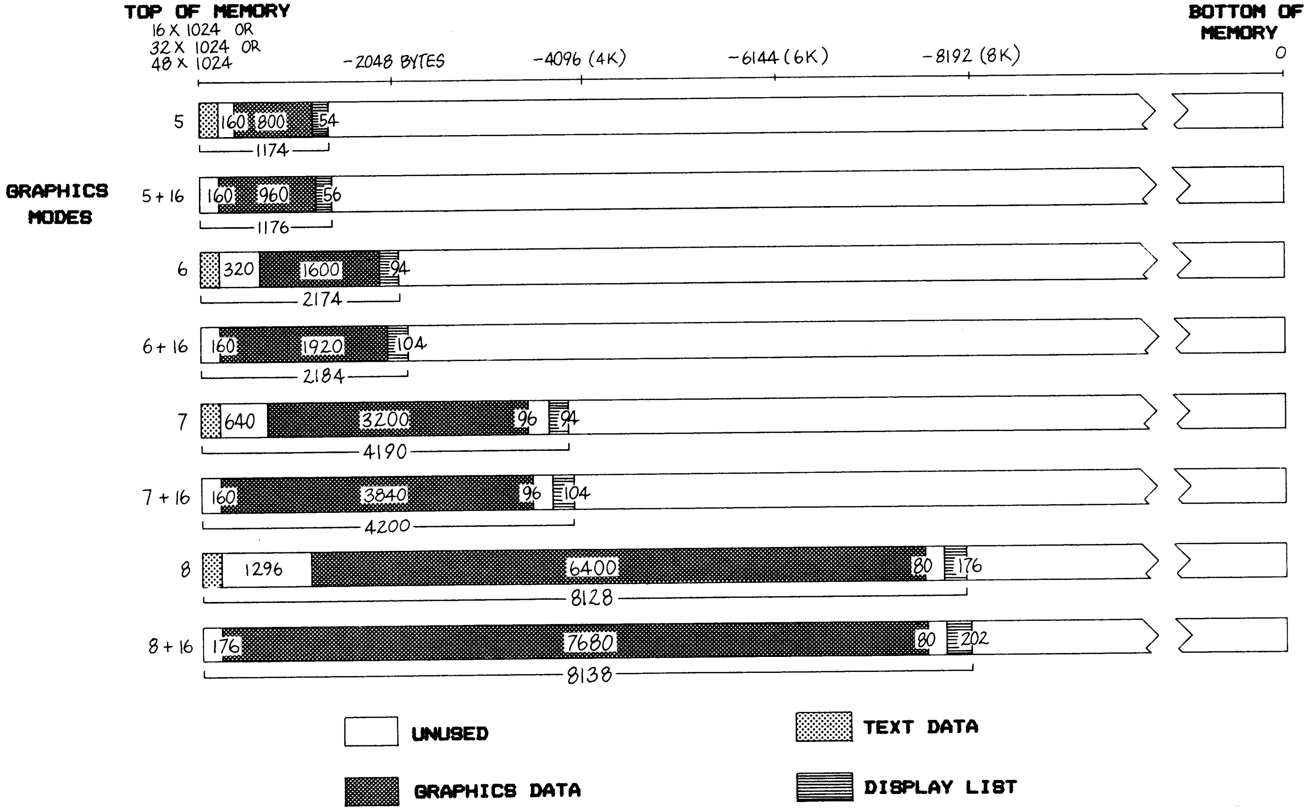


FIGURE 3B (NOT TO SCALE)

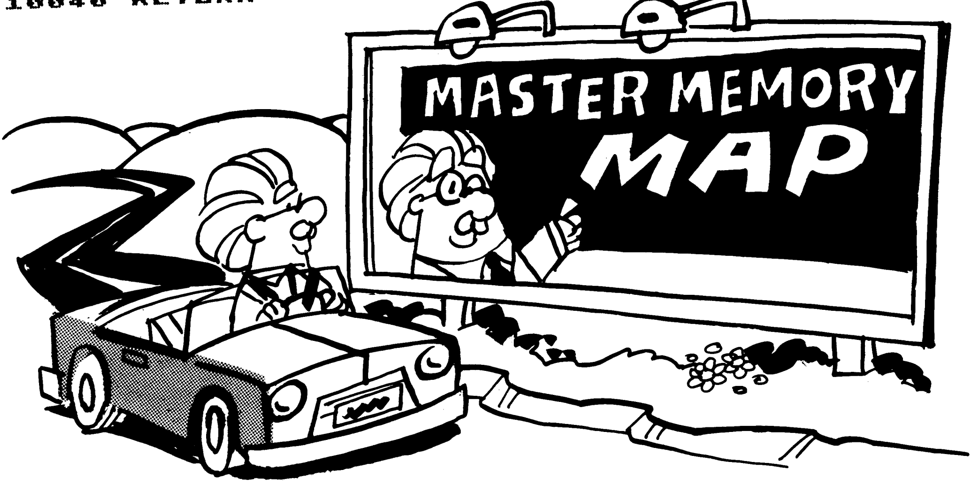
BACK TO THAT BIG CHART AGAIN

It isn't necessary for you to understand Figure 3, so feel free to skip this section. Just pick one of the Graphics Modes you are used to using and look at it on the chart. Lets pick GR.4 (O.S. mode 9). Starting down from the top of memory about 600 bytes, we see the DL uses 54 bytes of memory. Next comes the screen data of 400 bytes, then some empty (unused) memory locations. Finally, 160 bytes of data for the text window. The other Graphics Modes work the same way.

The text window is a slight complication to the DL after the sequence of 112's, the LMS bytes, and the various O.S. Mode numbers, as discussed above, YOU WOULD FIND ANOTHER LMS! And following that will come more mode line instructions (text of course), and now finally comes the 65 address to end the DL. The extra LMS just says to: "Go get some data from another location in memory than the one the computer is at, and then place that data in some text mode lines(usually 4 lines of GR.0), all at the bottom of the screen.

```
10 GRAPHICS 17:? #6;"  EXAMPLE 8"
20 FOR W=1 TO 230:POKE 710,W
30 FOR ZR=1 TO 10:NEXT ZR
40 NEXT W
500 REM THIS IS A SUBROUTINE FOR YOU TO USE I
N YOUR OWN PROGRAMS. SEE MANUAL FOR DIRECTION
S
700 GOSUB 10000
710 REM PUT THE REST OF YOUR PROGRAM STARTING
ON THIS LINE
715 FOR W=1 TO 2000:NEXT W
720 GRAPHICS 0:? "THIS EXAMPLE IS FOR YOU TO
USE IN YOUR PROGRAMS.":? "PRESS BREAK AND SAV
E IT IF YOU WANT OR";
```

```
730 ? " PRESS OPTION TO GO ON.":POKE 53279,8
740 Z=PEEK(53279)
750 IF Z=3 THEN POKE 764,12:RUN "D:EX4"
760 GOTO 740
10000 TRAP 10040
10010 DL=PEEK(560)+256*PEEK(561)
10011 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
10012 I=1
10013 READ A:IF A=0 THEN 10040
10014 POKE (DL+I-1),A
10015 IF I=5 THEN POKE DL+4,Z1
10016 IF I=6 THEN POKE DL+5,Z2
10020 I=I+1:GOTO 10013
10030 DATA 112,112,112,66,64,156,2,2,2,5,5,5,
5,2,2,4,4,4,4,4,2,65,32,124,0,0,0
10040 RETURN
```



NOW FOR AN ADVERTISEMENT

All these PEEK and POKE locations are summerized in my MASTER MEMORY MAP. FORK OVER THE $6.95 and order one today!

I hate ads too, but....News flash!!! The ATARI isn't as slow as some reviewers have stated in bench tests. The ATARI ANTIC chip that we have been programming steals time from the main 6502 processor. The bigger the DL and the more data it has to put on the screen, the more time it steals. This means that if a reviewer tested the machine in GR.0 it will run calculations slower than if in GR.2 or 3 which have smaller DL's. In fact, if you want you can customize a DL to have only blank lines with one small message in the middle of the screen that says "Wait a Moment", this will allow the machine to run almost as fast as it can go. In case you want to know the difference; in GR.8 the machine is up to 40% slower than GR.2!

EXAMPLE 4 & 5

Ok students, time for fun. Run example 4. All of this DL stuff may not seem too practical, so the next two examples are to give you some ideas. By simply setting up a GR.1 (large letters) DL, but in the middle doing a second LMS (70,96,127 in the data but the last two numbers are rePOKEd in lines 10023-10024 depending on your memory), we tell the DL that once it gets about halfway down the screen, it should get its data from the same place as the top of the screen did! There are few restrictions on what you can do with the LMS byte. It doesn't have to wait to be used only at the beginning of the DL and at 4K boundaries.

Example 5 is the same thing, but with only one copy of what you type in. Both of these could be used for games, but a more useful idea might be to allow programming in big letters for the hard of seeing or kids to use.

HAVE I LOST YOU?

You should still be in example 4. The program says READY in double letters. Since the program has stopped, you can do what you like. Try a simple line like "Hello, I use BIG letters". The computer will now work like it was in GR.0. Why? Look at line 720 in EX.4--720 POKE 87,0:END. The POKE to location 87 fools the computer. We just (in the subroutine) set up the DL like a GR.1 screen, but the computer looks to location 87 to see what mode it is in, so by POKEing in 0 (the Graphics Mode goes here, not the O.S. mode used in the DL), it thinks it can write listings as though it were in GR.0.

```
10 GRAPHICS 17:? #6;" EXAMPLE [illegible]"
20 FOR W=1 TO 230:POKE 710,W
30 FOR ZR=1 TO 10:NEXT ZR
40 NEXT W
490 REM ****************************
```

```
500 REM HERE IS A PRACTICAL EXAMPLE OFWHAT YO
U CAN DO WITH A DISPLAY LIST THAT IS MODIFIED
510 REM JUST SAY GOTO 700    WHEN YOU  WANT T
O SEE YOUR PROGRAM IN BIG  LETTERS AND IN DUP
LICATE!!
520 REM ******************************
700 GOSUB 10000
710 REM YOUR BASIC PROGRAM GOES HERE
720 POKE 87,0:END
10000 REM TRAP 10040:GRAPHICS 1
10005 GRAPHICS 1
10010 DL=PEEK(560)+256*PEEK(561)
10015 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
10016 I=1
10017 READ A:IF A=0 THEN 10040
10020 POKE (DL+I-1),A
10021 IF I=5 THEN POKE DL+4,Z1-32
10022 IF I=6 THEN POKE DL+5,Z2+2
10023 IF I=16 THEN POKE DL+15,Z1-32
10024 IF I=17 THEN POKE DL+16,Z2+2
10025 I=I+1:GOTO 10017
10030 DATA 112,112,112,70,96,127,6,6,6,6,6,6,
6,6,70,96,127,6,6,6,6,6,6,6,6,65,94,125,0,0
10040 RETURN
```

Example 5 is almost the same, except instead of the POKE 87, we start with a GR.0 mode. This internally changes 87 to 0 (try a PEEK(87) if you don't believe me). Sometimes one method is more desirable than the other. You pick one. The other difference is that there is no second LMS in the middle of the DL. The writing in these two examples is large, but the computer still thinks it is in mode 0 so it trys to fit in the normal 40x24 letters. For this reason, you won't see the text in the way that you expect. Try a LIST command. STRANGE!

NOTE - To get to example 5 & 6 you have to either type in RUN"D:EX5" OR RUN"C:", whichever is appropiate. The same for EX.6. This is because we stopped EX.4 & EX.5 for you to try out.

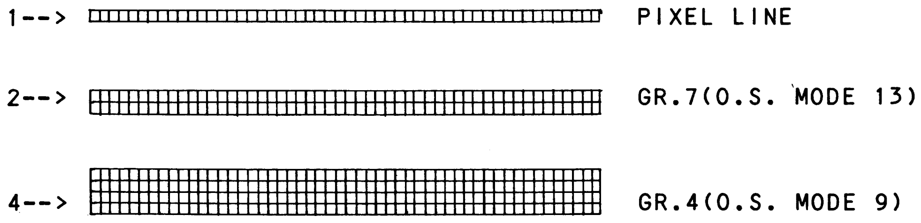
```
10 GRAPHICS 17:? #6;"  EXAMPLE 5"
20 FOR W=1 TO 230:POKE 710,W
30 FOR ZR=1 TO 10:NEXT ZR
40 NEXT W
500 REM THIS IS A SUBROUTINE TO ALLOW YOU TO
PROGRAM WITH BIG LETTERS.SEE MANUAL FOR DIRE
CTIONS.
700 GRAPHICS 0:GOSUB 10000
710 REM PUT THE REST OF YOUR PROGRAM  STARTIN
G ON THIS LINE
720 LIST :END
10000 TRAP 10040
10010 DL=PEEK(560)+256*PEEK(561)
10011 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
```

```
10012 I=1
10013 READ A:IF A=0 THEN 10040
10014 POKE (DL+I-1),A
10015 IF I=5 THEN POKE DL+4,Z1+70
10016 IF I=6 THEN POKE DL+5,Z2+2
10020 I=I+1:GOTO 10013
10030 DATA 112,112,112,70,74,158,6,6,6,6,6,6,
6,6,6,6,6,6,6,6,6,6,6,6,65,32,124,0,0,0
10040 RETURN
```

EXAMPLE 6

This example is like EX.2, except that it remembers your current DL. This way you can slowly change the DL until you like it. Then copy the numbers and use them in EX.3 in your own programs. After you create a screen, if you want to BREAK the program just type LIST. This will flow the programs data over the DL you just created to better see each region. To restart the program, just type RUN. Other commands are: press OPTION to go to EX.7; press START to do another screen using the current DL numbers; press SELECT to get back to the original numbers in case your custom screen goes crazy, or you get lost in what you're doing.

GRAPHICS MODE LINES AND PIXEL LINES



(Up to 16 Pixels per Mode Line in O.S. Modes 5 & 7 (=GR.2))

FIGURE 4

GOING CRAZY?

Yes, remember we said before that if you put in enough modes in the DL so that too many pixel lines are used up, the TV will start to roll. This is simply because you are saying in the DL to "Draw on the screen more than you have room for". Conversely, if you put too few lines on the screen, the picture will shrink. Simply change the amount of modes or the type of mode(ie., change several modes like 2 which use 8 lines of pixels per mode line of 2(GR.0), to ones like mode 9 which use only 4 pixel rows per mode 9(GR.4) line). Now,if this mode line vs. pixel line stuff has got you lost, we are going to have one more figure (figure 4).This one shows clearly (we hope) the difference between the two.

```
1 GRAPHICS 17:? #6;"  EXAMPLE 5"
3 FOR W=1 TO 230:POKE 710,W
4 FOR ZR=1 TO 10:NEXT ZR
5 NEXT W
7 DIM ZX$(135)
20 GRAPHICS 0:? "THIS EXAMPLE ALLOWS YOU TO C
HANGE A   MODE 0 DISPLAY LIST!"
 30 ? "PRESS START TO BEGIN":? "WRITE DOWN THE
  NUMBERS IN THE DL WHEN YOU LIKE IT!"
55 DATA 112,112,112,66,96,144,2,2,2,2,2,2,2,2
,2,4,4,4,4,4,2,2,2,2,2,2,2,2,2,65,32,156,0
56 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
57 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
58 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
62 POKE 53279,8
65 Z=PEEK(53279)
70 IF Z=6 THEN 100
80 GOTO 65
100 GRAPHICS 0:POSITION 2,3
105 LIST 55,58
110 POSITION 2,18:? "CHANGE THE VALUES YOU WA
NT BY USING   THE CURSOR CONTROLS "
120 POSITION 10,0
125 INPUT ZX$
130 POSITION 0,2
```

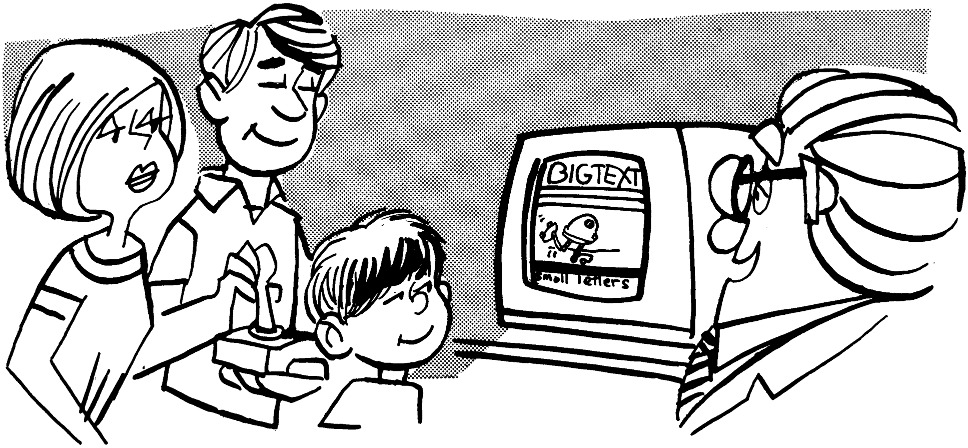
```
140 POKE 842,13
142 POSITION 2,17:? "CONT"
144 POSITION 0,2:STOP
150 POKE 842,12
200 RESTORE 55
210 I=0
215 TRAP 400
217 I=1:GRAPHICS 7+16
218 DL=PEEK(560)+256*PEEK(561)
219 A=PEEK(DL+4):B=PEEK(DL+5)
220 READ G:IF G=0 THEN 233
221 POKE DL+I-1,G
222 IF I=5 THEN POKE DL+4,A
223 IF I=6 THEN POKE DL+5,B
230 I=I+1:GOTO 220
233 FOR W=1 TO 900:NEXT W:POKE 87,0
240 POSITION 2,1:? "PRESS OPTION TO GO ON":?
"PRESS START TO DO ANOTHER SCREEN"
243 ? "PRESS SELECT FOR THE ORIGINAL DISPLAY
LIST"
245 TRAP 5
250 POKE 53279,8
260 Z=PEEK(53279)
265 IF Z=5 THEN 400
270 IF Z=6 THEN 57
280 IF Z=3 THEN POKE 764,12:RUN "D:EX7"
300 GOTO 260
400 TRAP 400:GRAPHICS 0:POSITION 2,4
405 ? "55 DATA 112,112,112,66,";A;",";B;",2,2
,2,2,2,2,2,2,2,4,4,4,4,4,2,2,2,2,2,2,2,2,2,65
,32,156,0"
406 ? "56 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0"
407 ? "57 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0"
408 ? "58 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0"
410 GOTO 110
```

EXAMPLE 7

This one is for you to figure out. Some clues: When the "HELLO'S" stop, look at all the modes on the screen. Notice towards the bottom, the mode eight area (obvious by the red and blue pixels). Since the computer displays the data according to whatever data is next "in line", the statements about press OPTION, START, or SELECT end up in this mode 8 area (even though you can't see the statements, you can still use them). Then we hit a 4k boundary, so a LMS instruction was used but to make it interesting, we used the same address as the top of the data. Look at the Display List for GR.8 using EX.1. See the 79 (LMS) instruction in the middle.

```
2 GRAPHICS 17:? #6;"  EXAMPLE 7"
4 FOR W=1 TO 230:POKE 710,W
5 FOR ZR=1 TO 10:NEXT ZR
6 NEXT W
7 DIM ZX$(135)
10 TRAP 40000:TRAP 10
20 GRAPHICS 0:? "THIS EXAMPLE ALLOWS YOU TO C
HANGE A   MODE 0 DISPLAY LIST!"
30 ? "PRESS START TO BEGIN":? "WRITE DOWN THE
NUMBERS IN THE DL WHEN YOU LIKE IT!"
40 POKE 710,0
55 DATA 112,112,112,66,96,144,2,2,2,2,6,6,7,7
,14,14,14,14,14,14,14,2,2,15,15
56 DATA 15,15,15,15,15,79,96,144,15,6,7,2,2,6
5,32,156,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65
,32,156,0
57 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
58 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
62 POKE 53279,8
65 Z=PEEK(53279)
70 IF Z=6 THEN 100
80 GOTO 65
100 ? "↰":POSITION 2,3
105 LIST 55,58
110 POSITION 2,18:? "CHANGE THE VALUES YOU WA
NT BY USING   THE CURSOR CONTROLS "
115 ? "SCREEN."
120 POSITION 10,0
125 INPUT ZX$
130 POSITION 0,2
140 POKE 842,13
142 POSITION 2,17:? "CONT"
144 POSITION 0,2:STOP
150 POKE 842,12
200 RESTORE 55
210 I=0
215 TRAP 400
217 I=1:GRAPHICS 7+16:POKE 710,0
218 DL=PEEK(560)+256*PEEK(561)
219 A=PEEK(DL+4):B=PEEK(DL+5)
220 READ G:IF G=0 THEN 233
221 POKE DL+I-1,G
222 IF I=5 THEN POKE DL+4,A
223 IF I=6 THEN POKE DL+5,B
230 I=I+1:GOTO 220
233 FOR W=1 TO 900:NEXT W:POKE 87,0:LIST :GOS
UB 5000
240 ? "PRESS OPTION TO GO ON":? "PRESS START
TO DO ANOTHER SCREEN"
243 ? "PRESS SELECT FOR THE ORIGINAL DISPLAY
LIST"
```

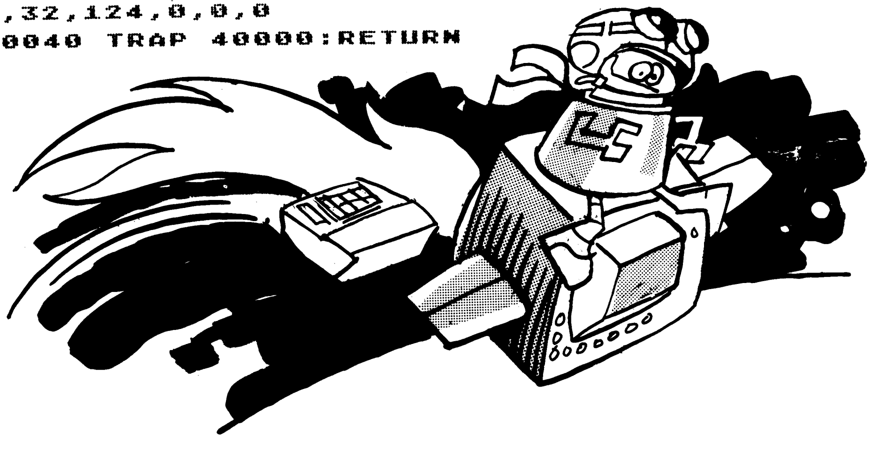
```
245 TRAP 10
250 POKE 53279,8:J=0
260 Z=PEEK(53279)
265 IF Z=5 THEN 400
270 IF Z=6 THEN 57
280 IF Z=3 THEN POKE 764,12:RUN "D:EX8"
300 GOTO 260
400 TRAP 40000:GRAPHICS 0:POSITION 2,4
405 ? "55 DATA 112,112,112,66,";A;",";B;",2,2
,2,2,6,6,7,7,14,14,14,14,14,14,14,2,2,15,15"
406 ? "56 DATA 15,15,15,15,15,79,96,144,15,6,
7,2,2,65,32,156,0,0,0,0,0,0,0,0,0"
407 ? "57 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0"
408 ? "58 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0"
410 GOTO 110
5000 J=J+1:? "HELLO":IF J<50 THEN 5000
5010 RETURN
```

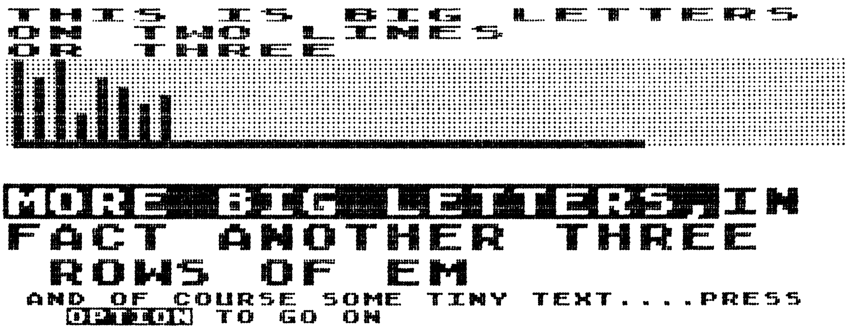


EXAMPLE 8

Get your wife, husband, kids, or friends. You are about to prove what home computers are good for. Remember that the LMS numbers (two numbers after the LMS value) tell the computer where to get the screen data. Well, this example just uses the joystick to change this address so you can see most of the memory in your machine in both graphics and text, some of it changing before your eyes. Moving the joystick forward will move up to the top of memory(remember you were almost at the top to start). Moving down will go almost to the beginning of memory. If we were to look at the very bottom we would crash(stop) the computer because we would interfere with the O.S. OK.....it wasn't that great....send them all back. Press OPTION to go on.

```
10 GRAPHICS 17:? #6;"  EXAMPLE 8"
20 FOR W=1 TO 230:POKE 710,W
30 FOR ZR=1 TO 10:NEXT ZR
40 NEXT W
500 REM THIS IS A SUBROUTINE FOR YOU TO USE T
O LOOK AT MEMORY BOTH FROM A GRAPHICS AND TEX
TUAL VIEWPOINT!
520 REM USE YOUR JOYSTICK AND MOVE UP OR DOWN
 IN MEMORY !!! PRESS OPTION TO GO ON.
700 GRAPHICS 0:GOSUB 10000
705 I=Z1:LIST :POKE 53279,8
707 FOR F=1 TO 50:? "USE JOYSTICK":NEXT F
710 ST=STICK(0):Z=PEEK(53279):IF Z=3 THEN POK
E 764,12:RUN "D:EX9"
715 IF ST=15 THEN 710
720 IF ST=14 THEN Z2=Z2+1
730 IF ST=13 THEN Z2=Z2-1
740 IF Z2<10 THEN Z2=10
745 IF Z2>PEEK(106) THEN Z2=PEEK(106)
750 POKE DL+5,Z2
760 GOTO 710
10000 TRAP 10040
10010 DL=PEEK(560)+256*PEEK(561)
10011 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
10012 I=1
10013 READ A:IF A=0 THEN 10040
10014 POKE (DL+I-1),A
10015 IF I=5 THEN POKE DL+4,Z1
10016 IF I=6 THEN POKE DL+5,Z2
10020 I=I+1:GOTO 10013
10030 DATA 112,112,112,66,64,156,2,2,2,2,2,2,
6,6,6,6,6,6,6,6,10,10,10,10,10,10,2,2,2,2,2,6
5,32,124,0,0,0
10040 TRAP 40000:RETURN
```





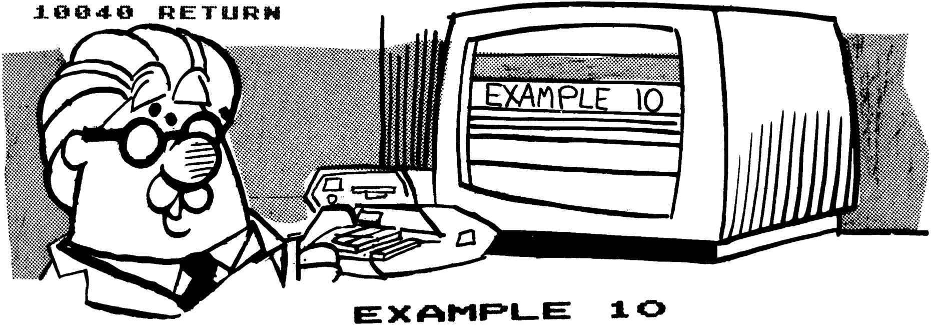
```
THIS IS BIG LETTERS
ON TWO LINES
OR THREE

MORE BIG LETTERS,IN
FACT ANOTHER THREE
 ROWS OF EM
 AND OF COURSE SOME TINY TEXT...PRESS
  OPTION TO GO ON
```

EXAMPLE 9
A PRACTICAL EXAMPLE

Finally an actual example where we PLOT and PRINT stuff. Now the problem with custom DL's is not choosing the numbers as you have already seen. That was easy. But now we have got to get the PLOT or PRINT statements to work with the new screen we have setup. When we start out a custom DL with a POKE 87,GR.mode # or a Graphics statement like GR.0, the O.S. thinks that it is in that mode, not the one on the screen. If the areas we are plotting or writing to are within the normal limits, then we can just experiment to find what mode line we want to PLOT/PRINT to. That is what this example does. It's hit or miss, but it works. Plot something and if it's not where you want change the PLOT statement. Same with text, using the POSITION statement. There is nothing wrong with the "hit and miss" method, but our next example will give you the precise numbers to accurately place your information on the screen.

```
10 GRAPHICS 17:? #6;"   EXAMPLE 9"
20 FOR W=1 TO 230:POKE 710,W
30 FOR ZR=1 TO 10:NEXT ZR
40 NEXT W
500 REM THIS IS A SUBROUTINE FOR YOU TO USE I
N YOUR OWN PROGRAMS. SEE MANUAL FOR DIRECTION
S.
505 GRAPHICS 0
510 GOSUB 10000
515 POKE 87,1:POSITION 0,0:? #6;"THIS IS BIG
LETTERS"
520 ? #6;"ON TWO LINES":? #6;"OR THREE"
550 POKE 87,4:COLOR 1:PLOT 1,5:DRAWTO 1,15:DR
AWTO 60,15:PLOT 15,15:DRAWTO 15.10:PLOT 3,15:
DRAWTO 3,8
555 PLOT 5,15:DRAWTO 5,6:PLOT 7,15:DRAWTO 7,1
2:PLOT 9,15:DRAWTO 9,8:PLOT 11,15:DRAWTO 11,9
:PLOT 13,15:DRAWTO 13,11
560 POKE 87,2:POSITION 0,9:? #6;"MORE BIG LET
TERS,in fact an other three   rows of em"
570 POKE 87,0:POSITION 1,6:? "AND OF COURSE S
OME TINY TEXT...PRESS   OPTION TO GO ON"
```

```
730 POKE 53279,8
740 Z=PEEK(53279)
750 IF Z=3 THEN POKE 764,12:RUN "D:EX10"
760 GOTO 740
10000 TRAP 10040
10010 DL=PEEK(560)+256*PEEK(561)
10011 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
10012 I=1
10013 READ A:IF A=0 THEN 10040
10014 POKE (DL+I-1),A
10015 IF I=5 THEN POKE DL+4,Z1
10016 IF I=6 THEN POKE DL+5,Z2
10020 I=I+1:GOTO 10013
10030 DATA 112,112,112,70,64,156,6,6,9,9,9,9,
9,9,9,9,9,9,7,7,7,7,2,2,2,2,2,2,65,32,124,0,0
,0
 10040 RETURN
```



EXAMPLE 10

This example is the big time! We have placed all of the previous abilities in this one, but added the information we promised in a chart on the screen. RUN the example and follow it through with us.

The program still uses DATA statements to experiment with breaking up the screen into mixed graphic and text modes. Use the cursor controls as before to edit the list, press RETURN to see what you have created. If you used PLOT modes at the top of the screen, you won't see that the controls are different, so pay attention. When the split screen appears, you can :

1) Press START (as before) to go change the current DL.
2) Press SELECT (as before) to go back to the original DL.
3) Press OPTION and START at exactly the same time. This will get you to the last example.
4) Press OPTION for a chart of the custom DL you created.

Now get to the new chart by pressing RETURN and then OPTION as explained above. The first column is the position of each graphics mode you choose:1,2,3...last(maximum of 20!). The second column tells you what number to POKE into location 87. This is simply the regular graphic mode number(0-11). Column three is the O.S.mode # which was the number you used in the DL itself. Yes, it is confusing to use both numbers in the same program when they mean the same thing to us, but remember that when you use a regular GR. mode # you are talking to the BASIC interpeter which then changes the number you give into the O.S. mode number . We are just bypassing the interpeter to get the special screens that it doesn't allow!

The next column is the number of lines of this mode you asked for, ie. how many times you included it in the DL. This is useful because if you have say six lines of a graphic region and you try to plot to 7,7 you will go out of that mode and into the next, causing garbage to appear on the screen. The same goes for text regions. This column will provide a quick reference. Then comes the two columns of POKE numbers for locations 88 & 89. These two values combined with 87 will fool the computer into acting as if each region starts with 0,0 at it's upper left corner. This makes your PRINT and PLOT statements work like normal. The difference is you PRINT/PLOT to EACH region as if it started with 0,0 at it's upper left corner.

Write down all this information or use in your custom routine, or to modify one of our examples.

The chart also lists the number of lines of pixels you filled on the screen. If it doesn't come out close to 192, or if you want to make some other change, you can go back to re-edit the DL by pressing START. If you have too many lines, then remove modes lines, or add some if you're short.

IF YOU'RE STILL LOST

Use the parts of this program that you understand now, and come back to study and experiment with the rest of it later. Basically, you should eventually understand the following to use this program. All the rest is "additional information":

1) Use EX.10 to get;
    a)the DL to look about right.
    b)the numbers to POKE into 87, 88, and 89.

2) Take EX.11 and change the POKE values to what EX.10 will give you when you tranfer the numbers from EX.9 into the DL data of EX.10.
3) Try to change the graphics and text we used to your own.
4) If you have more regions than the example has, add more. Likewise if you have less.
5) Remember to treat each area on the screen like it was a small screen by itself... don't PLOT or PRINT too far.

```
1 GRAPHICS 17:? #6;"  EXAMPLE 10"
2 FOR W=1 TO 230:POKE 710,W
3 FOR ZR=1 TO 10:NEXT ZR
4 NEXT W
5 DIM P(220):DIM NUM(20):DIM MODE(20):DIM RAM
(15):DIM START(20):DIM P88(20):DIM P89(20):DI
M LIN(15)
7 DIM BAM(15)
10 TRAP 490
20 DIM ZX$(135)
30 GRAPHICS 0:? "THIS EXAMPLE ALLOWS YOU TO C
HANGE A   MODE 0 DISPLAY LIST!"
40 ? "PRESS START TO BEGIN.":? "WRITE DOWN TH
E NUMBERS IN THE DL WHEN YOU LIKE THEM!"
50 DATA 112,112,112,66,64,156,2,2,2,2,2,2,2,2
,2,4,4,4,4,4,2,2,2,2,2,2,2,2,2,65,32,156,0
60 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
70 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
80 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0
90 POKE 53279,8
100 Z=PEEK(53279)
110 IF Z=6 THEN 140
120 GOTO 100
130 TRAP 490
140 GRAPHICS 0:POSITION 2,3
150 LIST 50,80
160 POSITION 2,18:? "CHANGE THE VALUES YOU WA
NT BY USING   THE CURSOR CONTROLS."
170 ? "PRESS RETURN TO SEE THE NEW SCREEN."
180 POSITION 10,0
190 INPUT ZX$
200 POSITION 0,2
210 POKE 842,13
220 POSITION 2,17:? "CONT"
230 POSITION 0,2:STOP
240 POKE 842,12
```

```
250 RESTORE 50:MAX=0
260 TRAP 490:READ S:IF S>15 THEN 260
265 IF S>MAX THEN MAX=S
270 IF S=0 THEN 281
280 GOTO 260
281 ON MAX GOTO 285,285,285,285,285,285,285,2
85,285,285,286,287,287,288,288
285 USE=0:GOTO 290
286 USE=22:GOTO 290
287 USE=23:GOTO 290
288 USE=24
290 RESTORE 50
300 TRAP 490
310 I=1:GRAPHICS USE
320 DL=PEEK(560)+256*PEEK(561)
330 A=PEEK(DL+4):B=PEEK(DL+5)
340 READ G:IF G=0 THEN 390
350 POKE DL+I-1,G
360 IF I=5 THEN POKE DL+4,A
370 IF I=6 THEN POKE DL+5,B
380 I=I+1:GOTO 340
390 FOR Q=1 TO 100:NEXT Q:POKE 87,0
400 POSITION 2,1:? "PRESS OPTION FOR CHART OF
PLOT VALUES WHEN DONE."
410 ? "PRESS SELECT FOR THE ORIGINAL DISPLAY
LIST #'S."
420 ? "PRESS START TO CHANGE CURRENT DL
NUMBERS."
425 ? "PRESS START & OPTION TO RUN LAST
EXAMPLE."
430 POKE 53279,8
440 Z=PEEK(53279)
450 IF Z=5 THEN 490
460 IF Z=6 THEN 70
465 IF Z=2 THEN RUN "D:EX11"
470 IF Z=3 THEN 550
480 GOTO 440
490 TRAP 40000:GRAPHICS 0:POSITION 2,4
500 ? "50 DATA 112,112,112,66,"";A;"",""";B;"",2,2
,2,2,2,2,2,2,2,4,4,4,4,4,2,2,2,2,2,2,2,2,2,65
,32,156,0"
510 ? "60 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,65,32,156,0"
520 ? "70 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0 ,0,0,0,0,0,0,0,0,65,32,156,0"
 530 ? "80 DATA 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
 0,0,0,0,0,0,0 ,0,0,0,0,0,0,0,0,65,32,156,0"
 540 GOTO 160
 550 RESTORE 50:POKE 559,0
 560 J=1:K=1:L=2:P(1)=PEEK(DL+3)-64
 570 READ PR:P(L)=PR:IF P(L)>15 THEN 570
 580 IF P(L)=0 THEN 620
```

```
590 IF P(L)<>P(L-1) THEN MODE(K)=P(L-1):NUM(K
)=J:J=1:L=L+1:K=K+1:GOTO 570
600 J=J+1:L=L+1:GOTO 570
620 MODE(K)=P(L-1):NUM(K)=J
630 RAM(2)=40:RAM(3)=40:RAM(4)=40:RAM(5)=40:R
AM(13)=40:RAM(14)=40:RAM(15)=40
633 LIN(2)=8:LIN(3)=10:LIN(4)=8:LIN(5)=16:LIN
(13)=2:LIN(14)= 1:LIN(15)=1
634 LIN(6)=8:LIN(7)=16:LIN(8)=8:LIN(9)=4:LIN(
10)=4:LIN(11)=2:LIN(12)=1
635 RAM(6)=20:RAM(7)=20:RAM(8)=10:RAM(9)=10:R
AM(10)=20:RAM(11)=20:RAM(12)=20
640 FOR H=1 TO K
644 BAM(2)=0:BAM(3)=40:BAM(4)=40:BAM(5)=40:BA
M(13)=7:BAM(14)=40:BAM(15)=8
645 BAM(6)=1:BAM(7)=2:BAM(8)=3:BAM(9)=4:BAM(1
0)=5:BAM(11)=6:BAM(12)=40
660 IF MODE(H)<MODE(H-1) THEN 680
670 SAV=MODE(H)
680 NEXT H
690 IF SAV>6 THEN 720
700 IF SAV>0 THEN SAV=6:GOTO 720
710 IF SAV<>0 THEN ? "ERROR AT 710..SAV SHOUL
D BE 0":STOP
720 MAXRAM=RAM(SAV)
725 FOR M=1 TO K:M2=0:M3=0
735 FOR S=1 TO M-1
740 IF RAM(MODE(S))=20 THEN 760
750 IF RAM(MODE(S))=10 THEN 770
755 GOTO 780
760 M2=NUM(S)+M2:GOTO 780
770 M3=NUM(S)+M3
780 NEXT S
783 R=0
785 FOR W=1 TO M-1:R=R+NUM(W):NEXT W
790 CHRPOS=A+B*256+(R*MAXRAM-M2*(MAXRAM-20)-M
3*(MAXRAM-10))
795 IF M=1 THEN CHRPOS=A+B*256
800 START(M)=CHRPOS
810 NEXT M
820 FOR Z=1 TO K:P89(Z)=INT(START(Z)/256):P88
(Z)=START(Z)-P89(Z)*256:NEXT Z
825 SUM=0
830 FOR Q=1 TO K:SUM=NUM(Q)*LIN(MODE(Q))+SUM:
NEXT Q
840 GRAPHICS 0:POSITION 0,0:? "┌─────────────
───────────────────────┐"
850 POSITION 0,1:? "|POS|POKE|O.S. |# LINES|
POKE | POKE |"
860 POSITION 0,2:? "|ON |INTO|MODE |OF THIS|
INTO | INTO |"
```

```
870 POSITION 0,3:? "|SCR| 87 | #    | MODE   |
 88  |  89  |"
880 POSITION 0,4:? "└──────────────────────────────────────
──────────────┘"
890 FOR W=1 TO K
895 SD=BAM(MODE(W))
898 POSITION 1,4+W:? W;
901 IF SD=40 THEN 903
902 POSITION 6,4+W:? SD;:GOTO 904
903 POSITION 6,4+W:? "NONE";
904 POSITION 12,4+W:? MODE(W):POSITION 18,4+W
:? NUM(W):POSITION 25,4+W:? P88(W);
905 POSITION 33,4+W:? P89(W)
910 NEXT W:? "THE NUMBER OF PIXEL LINES USED=
";SUM;".":? "SUGGEST YOU CHANGE DL UNTIL 192
PIXEL LINES RESULTS!";
915 ? " COPY NUMBERS OR PRESS  START TO REDO
CURRENT DL.":? "GRAPHICS MODE TO USE IN EX.11
 IS ";USE
 920 POKE 559,34
 930 POKE 53279,8
 940 Z=PEEK(53279)
 950 IF Z=6 THEN 140
 960 GOTO 940
 999 END
```

EXAMPLE 11
LAST EXAMPLE

This PLOT is just like EX.9, but now we add POKEs to 88 and 89 to fool the computer into acting as if the upper corner of each mode was position 0,0. The POKE to 87 then says "act as if you're in a regular graphics mode #" (your input). Now we just PLOT or PRINT as if each section was a small screen with 0,0 at the top like normal. If you PLOT too far, the next region down will look funny. We include the code for this example (minus a few lines you don't need) since this example is all you really need to understand. Our other examples will do all the calculating for you. Make sure you change the 5th and 6th numbers in the data list to the correct values for your computer, as explained above.

```
10 GRAPHICS 17:? #6;"  EXAMPLE 11"
20 FOR W=1 TO 230:POKE 710,W
30 FOR ZR=1 TO 10:NEXT ZR
40 NEXT W
50 ? #6;"  LaST ExAMPlE!"
60 FOR W=1 TO 230:POKE 708,W:POKE 709,W:POKE
710,W:POKE 711,W
```

```
65 FOR ZR=1 TO 5:NEXT ZR
70 NEXT W
80 GRAPHICS 17:? #6;"TO MAKE THIS EXAMPLE"
90 ? #6;"WORK, YOU WILL HAVE "
100 ? #6;"TO CHANGE THE VALUES"
110 ? #6;"EACH TIME POKE 88 OR"
120 ? #6;"89 IS USED. THIS IS "
130 ? #6;"DONE BY INPUTING THE"
140 ? #6;"VALUES IN THE DL OF "
150 ? #6;"THIS EX. INTO [illegible]."
160 ? #6;"the chart will give "
170 ? #6;"you the values for  "
180 ? #6;"each mode area.break"
190 ? #6;"[illegible]"
200 POKE 764,255
210 IF PEEK(764)<>12 THEN 210
400 REM ********************************
500 REM THIS IS A SUBROUTINE FOR YOU TO USE A
S A PRACTICE EXAMPLE IN PLOTTING AND WRITING
TO THE AREAS OF THE
503 REM SCREEN THAT YOU SET UP IN THE PREVIOU
S EXAMPLE. JUST PLUG IN THE POKE VALUES FOR L
OCATIONS 87,88,&89
504 REM AND CHANGE THE NUMBERS IN THE DATA ST
ATEMENT TO THOSE YOU WROTE DOWN FROM THE LAST
EXAMPLE . FINALLY,YOU
505 REM SHOULD CHANGE THE PLOT AND PRINT STAT
EMENTS TO THE ONES YOU WANT. HAVE FUN...SEE Y
OU IN THE NEXT TUTORIAL.
506 REM ********************************
507 GRAPHICS 0:REM PUT [illegible] HERE, REMEMBER TO
USE THE GRAPHICS MODE THAT THE CHART GAVE YO
U.
510 GOSUB 10000
515 POKE 87,1:POKE 88,64:POKE 89,156:POSITION
 0,0:? #6;"THIS IS BIG LETTERS"
520 ? #6;"ON TWO LINES":? #6;"OR THREE"
540 POKE 88,124:POKE 89,156
550 POKE 87,4:COLOR 1:PLOT 1,0:DRAWTO 1,10:DR
AWTO 60,10:PLOT 15,10:DRAWTO 15,5:PLOT 3,10:D
RAWTO 3,3
555 PLOT 5,10:DRAWTO 5,1:PLOT 7,10:DRAWTO 7,7
:PLOT 9,10:DRAWTO 9,3:PLOT 11,10:DRAWTO 11,4
:PLOT 13,10:DRAWTO 13,6
557 POKE 88,234:POKE 89,156
560 POKE 87,2:POSITION 0,0:? #6;"MORE BIG LET
TERS in fact another three    rows of em"
565 POKE 88,58:POKE 89,157
570 POKE 87,0:POSITION 0,0:? "AND OF COURSE S
OME TINY TEXT....PRESS   RETURN TO GO ON"
730 POKE 53279,8
740 Z=PEEK(53279)
750 IF Z=6 THEN 20000
```

```
760 GOTO 740
10000 TRAP 10040
10010 DL=PEEK(560)+256*PEEK(561)
10011 Z1=PEEK(DL+4):Z2=PEEK(DL+5)
10012 I=1
10013 READ A:IF A=0 THEN 10040
10014 POKE (DL+I-1),A
10015 IF I=5 THEN POKE DL+4,Z1
10016 IF I=6 THEN POKE DL+5,Z2
10020 I=I+1:GOTO 10013
10030 DATA 112,112,112,70,64,156,6,6,9,9,9,9,
9,9,9,9,9,9,9,7,7,7,7,2,2,2,2,2,2,2,65,32,124
,0,0,0
10040 RETURN
20000 GRAPHICS 0:POKE 710,0
20001 ? " THANKS FOR BUYING FROM SANTA CRUZ E
DUCATIONAL SOFTWARE!!!";:GOTO 20001
```

Final Notes

Sometimes when you do your custom screens you probably will have your text or plot suddenly shift over like this

to the other side of the screen. This is because your Display List is going along at, say, modes using 40 bytes per line when you shift and use three lines of a mode using 20 bytes per mode. The system is still trying to place the data where it used to go, so if you use 40's then switch to 20's or 10's, keep it in increments of the largest value. For example, if you use 40 byte modes then if you have 10 byte modes either use 4, 8, 12 , etc., so that the computer can find 40 at a time. DON'T FOOL MOTHER ATARI!

We also want to warn you that EX.10 occasionally can be fooled by unusual inputs, so if the chart seems incorrect it may be. This is not a serious problem because you are going to learn to do all the calculations yourself, AREN'T YOU???

The other modes in the chart (FIG.1) that aren't explained are for things like lowercase descenders and multi-colored letters. See, I told you not to ask. These would be good topics for another Tricky Tutorial(tm). Well, maybe SOMEDAY.....

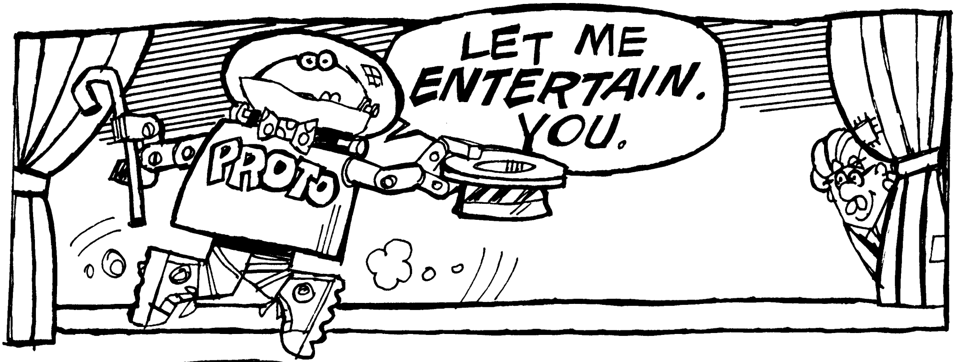
Well that's it. If even a new programmer will experiment with these examples, he or she will at least be able to make some nice custom screens without having to understand it all.

ALSO FROM EDUCATIONAL SOFTWARE

PROTOTYPE'S ADVENTURES

Ages 4 to 8

by Stan Gilbert. A delightful collection of 8 NON-VIOLENT and NON-COMPETITIVE games written especially for young children. Professor Von Chip's robot, Proto Type, will guide your child through activities such as catching free MARSHMALLOWS from friendly aliens, electronic COLORING, JUMP ROPE, and a fun game called ALPHA BLOCKS. Proto even shows them where he lives and works. The storybook, shaped like Proto Type, has cartoons for your child to color that don't need a computer to be fun. 16K-Tape/24K-Disk -- $19.95



DOG DAZE

by Gray Chang. A fast action arcade style game where the players are puppies fighting for control of the local fire hydrants. Each pooch races to claim the hydrants either by touching it first or by shooting his bone at it. And doglike, if a dog comes too close to another dog's hydrant, he's compelled to stop and sniff, losing valuable time. Each dog must be careful of cars or the game could end rather suddenly! Great fun for all ages! Written in efficient machine language. 8K-Tape/24K-Disk -- $16.95

SPACE GAMES

by Randy Massey. Three exciting games for one very affordable price. With all the fun you'll have playing ALIEN, SURVIVE, and ROBOT ATTACK you can't lose even if you get turned into space dust! 16K-Tape/32K-Disk -- $24.95 for all three.

IN THE UNLIKELY EVENT
YOU RECEIVE A DEFECTIVE PRODUCT
RETURN IT TO US FOR A
PROMPT REPLACEMENT OR REFUND

DISPLAY LISTS

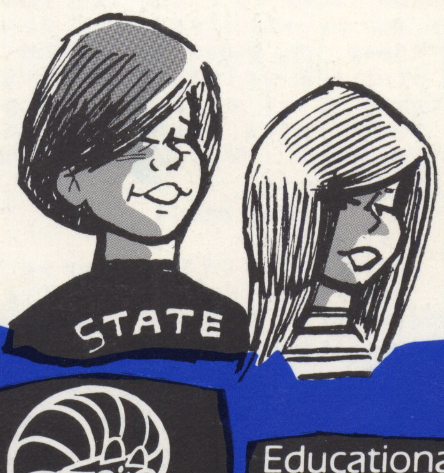
TRICKY TUTORIAL #1

Display Lists consists of a set of programs which instruct how to modify the ATARI's Display Lists to get multiple graphics modes on the screen at the same time. Using the examples and manual inside, you have only to follow a few simple directions to create your own custom screens. These screens can consist of any of the ATARI's regular Text and Graphics modes plus 5 new ones. Imagine up to 20 modes on the screen at once . . . . real special effects in all your programs!

This program requires very little actual programming experience. It is especially designed to allow you to use it now, and go back and learn the actual method within the program at any time in the future.

REQUIRES:

16K Tape / 24K Disk
BASIC Cartridge
1 Joystick



Educational
Software inc.

4565 Cherryvale Ave
Soquel, Ca. 95073
(408) 476-4901